ナップサック問題と劣モジュラ関数最大化問題の合意解形成による要約

安田 宜仁 † 西野 正彬 † 平尾 努 ‡ 鈴木 潤 ‡
† NTT サイバーソリューション研究所 ‡ NTT コミュニケーション科学基礎研究所
{yasuda.n,nishino.masaaki,hirao.tsutomu,suzuki.jun}@lab.ntt.co.jp

概 要

要約には，文字数制限という強い制約があるため限られた文字数に収まるよう重要な情報を重複なく含めなければならない．近年，この問題を最大被覆問題として解くことが主流となりつつある．しかし，最大被覆問題は NP 困難であるため，計算量が大きいという問題がある．本稿では，最大被覆問題のそれよりも現実的な計算量で解くための方略を提案する．重要情報を限られた文字数以内で最大限含めるという目的をナップサック問題として，情報の重複を最小化するという目的を劣モジュラ関数最大化問題として捉え，これらの部分問題の解の合意を取ることで最終的な要約を生成する．具体的にはラグランジュ緩和を利用して合意解を得るが，異なる目的で生成した要約の合意解形成は困難であるという問題がある．そこで，これを解決するさらなる緩和手法を提案する．

1　はじめに

要約は，限られた文字数内で元文書の情報を重複なく最大限含めることが求められる．近年，この問題は最大被覆問題として定式化されることが多い．つまり，与えられた文書 (あるいは文書集合) 中の概念 (ユニグラムやバイグラムなど) を，ある文字数内に収まるよう，出来る限り多く被覆するように文を選択することを目的とする．重複なく情報を網羅するという考え方は，Maxmum Marginal Relevance (MMR) [Goldstein 00] などの既存の要約手法の考え方と同等である．最初に要約生成を最大被覆問題とみなしたのは Filatova らであるが，彼らは貪欲法による解法しか用いておらず厳密解が得られる保証がない [Filatova 04]．その後，Takamura らによってより様々な解法が提案され，厳密解を得るアルゴリズムも提案された [Takamura 09]．しかし，厳密解を得るためのアルゴリズムは NP 困難であり，計算量が常に問題となる．特に複数文書要約のように要約の候補となる文の数が多い場合にはそれが深刻である．

本稿ではこの問題を解決するため，要約を

1. 疑似多項式時間で厳密解を得ることのできるナップサック問題，
2. 近似が必要な劣モジュラ関数最大化問題

に分解し，それらが出力した解の間の合意を取ることで効率的に要約を生成する手法を提案する．合意を取る手段としてラグランジュ緩和を導入するが，単純に用いるだけでは合意解を得ることが困難なことを示し，これを解決するヒューリスティクとして，長さ制限の段階的緩和を提案する．

TSC (Text Summarization Challenge) 3 [Hirao 04] のデータを用いて評価実験を行ったところ，個別に出力した解の間の合意を取ることで，合意を取らない場合と比較して ROUGE スコアが向上することを確認した．また，長さ制限の段階的緩和を導入することで，合意解の収束性が向上することも確認した．

2　関連研究

複数の文書から要約を生成 (以降，本稿では要約生成は文抽出により要約を生成することとする) する際に注意すべき点は，要約候補の文の間に冗長性，つまり，よく似た情報を共有する文があるということである．冗長性を制御して要約を生成する手法として，Goldstein らは MMR [Goldstein 00] を提案した．MMR では，要約として選ぶ文を逐次的に選択していく際，既に要約として選択した文と似た文には重要度にペナルティを与えることで冗長性を制御している．しかし，この手法では最終的な要約 (文集合全体) の冗長性を制御できない．McDonald らはこれをナップサック問題的に動的計画法で解く手法を提案した [McDonald 07]．冗

Algorithm 1 ナップサック問題としての要約

for $i \in \{0, \ldots, N\}$ **do**
  $C[i][0] \leftarrow 0$
for $\ell \in \{0, \ldots, L_{max}\}$ **do**
  $C[0][\ell] \leftarrow 0$
for $i \in \{1, \ldots, N\}$ **do**
  for $\ell \in \{1, \ldots, L_{max}\}$ **do**
    if $\mathbf{length}(s_i) \leq \ell$
      if $C[i \quad 1][\ell \quad \mathbf{length}(s_i)] + w(s_i) > C[i \quad 1][\ell]$
        $C[i][\ell] \leftarrow C[i \quad 1][\ell \quad \mathbf{length}(s_i)] + w(s_i)$
      else
        $C[i][\ell] \leftarrow C[i \quad 1][\ell]$
    else
      $C[i][\ell] \leftarrow C[i \quad 1][\ell]$

長性を考慮する必要がないのであれば，この手法で大域最適解を得ることができる．しかし，冗長性をペナルティとして用いると正確なナップサック問題にはならず，動的計画法では最適解を得ることができない．

最大被覆問題 [Filatova 04, Takamura 09] は文単位での冗長性を制御するのではなく，あらかじめ決定しておいた概念 (ユニグラムやバイグラムのような文内の小さな単位) を制限文字数以内で最大限被覆するように文を選ぶ．こうすることで自然と要約として情報の冗長性を削減することができるため，理にかなっている．しかし，この問題は NP 困難であり厳密解を得るためには計算量を無視できない．

Lin らは，ある文字数制約のもと，冗長性を排除しつつ概念を最大限被覆するという問題が劣モジュラ関数の最大化問題であることを示し[1]，貪欲法による効率的なアルゴリズムを示した [Lin 10, Lin 11]．アルゴリズムの解は，最悪でも最適解の $\frac{e-1}{e}$ を達成する．ただし，この手法では要約対象となる文集合をあらかじめクラスタリングしておかなければならず，そのための計算コストが余計にかかる．

一方，構文解析，機械翻訳といった他の自然言語処理分野では，こうした計算困難な問題を部分問題に分解し，部分問題が出力した解の間で合意を取ることによって最適解を効率的に求める手法が既に提案されている [Rush 10, Chang 11]．具体的には元となる最適化問題に対し，部分問題の間にラグランジュ緩和を導入することで解の合意を取る．

3 合意解形成による要約モデル

本稿では，重要な情報を要約に含めるという問題と冗長な情報を削減するという問題に対し，それぞれ，

[1] 文献 [Takamura 09] でも陽にではないが同様のことは述べられている．

Algorithm 2 劣モジュラ最大化問題としての要約

Input: $D = \{s_1, ..., s_N\}, L_{max}$
Initialize: $\boldsymbol{z} \leftarrow \mathbf{0}, \quad \ell \leftarrow 0, \quad C \leftarrow \{1, \ldots, N\}$
while $C \neq \emptyset$ **do**
  $maxscore \leftarrow 0$
  $i_{max} \leftarrow \quad 1$
  for $i \in C$ **do**
    $\boldsymbol{t} \leftarrow \boldsymbol{z}$
    $t[i] \leftarrow 1$
    $score \leftarrow \dfrac{\text{num\_bigrams}(\boldsymbol{t}) \quad \text{num\_bigrams}(\boldsymbol{z})}{\mathbf{length}(s_i)}$
    if $score > maxscore$
      $maxscore \leftarrow score$
      $i_{max} \leftarrow i$
  if $i_{max} < 0$
    break
  if $\ell + \mathbf{length}(s_{i_{max}}) \leq L_{max}$
    $z[i_{max}] \leftarrow 1$
    $\ell \leftarrow \ell + \mathbf{length}(s_{i_{max}})$
  $C \leftarrow C \setminus i_{max}$
return $\boldsymbol{z}$

厳密解を得ることのできるナップサック問題と近似が必要な劣モジュラ関数最大化問題として捉え，これらの間で合意を形成する要約モデルを提案する．

3.1 ナップサック問題としての要約

文の重要度 $w(s_i)$ が与えられたとき，最大要約文字数 L_{max} 以下で文重要度の和を最大とする文の組合せを選ぶ問題は整数重みのナップサック問題として定式化できる．よって，その厳密解は動的計画法により得ることができる．Algorithm 1 にそのアルゴリズムを示す．$C[i][\ell]$ は，i 番目の文までを要約候補とした時の文字数 ℓ 以下で文重要度の総和の最大を表す．$C[N][L_{\max}]$ を求めた後，それを記録した文をバックトラックすると最適解としての文集合 (要約) が求まる．なお，文の重要度 $w(s_i)$ は文 s_i に含まれる単語重要度の和として以下の式で決定する．

$$w(s_i) = \sum_{t \in s_i} \text{tf}\cdot\text{idf}(t)$$

3.2 劣モジュラ最大化問題としての要約

冗長性が少ない要約を生成することを異なるバイグラムをなるべく多く要約に含める，つまり，ある文字数制限のもと異なりバイグラム数を最大化する問題と考える．この問題は NP 困難であるが，異なりバイグラム数を評価する関数が劣モジュラ関数であることに注意すると Lin [Lin 11] らと同様に最悪 $\frac{e-1}{e}$ で最適

Algorithm 3 要約生成アルゴリズム

Input:
  $D = \{s_1, ..., s_N\}$, L_{max}
Initialize:
  $u_i^{(0)} \leftarrow 0$ **for** $i \in \{1, \ldots, N\}$
  $L'_{max} \leftarrow L_{max}$
for $k \in \{1, \ldots, K\}$ **do**
  $\boldsymbol{y}^{(k)} \leftarrow \underset{\boldsymbol{y} \in \mathcal{Y}}{\text{argmax}}(f_{\text{knp}}(\boldsymbol{y}) + \sum_{i=1}^{N} u_i^{(k-1)} y_i)$
  $\boldsymbol{z}^{(k)} \leftarrow \underset{\boldsymbol{z} \in \mathcal{Z}}{\text{argmax}}(f_{\text{grd}}(\boldsymbol{z}) \quad \sum_{i=1}^{N} u_i^{(k-1)} z_i)$
  if $y_i^{(k)} = z_i^{(k)}$ **for all** $i \in \{1, \ldots, N\}$
    return $\boldsymbol{y}^{(k)}$
  else
    for $i \in \{1, \ldots, N\}$ **do**
      $u_i^{(k)} \leftarrow u_i^{(k-1)} \quad \alpha^{(k)}(y_i^{(k)} \quad z_i^{(k)})$
return $\boldsymbol{y}^{(K)}$

解を得ることができる．Algorithm 2 に疑似コードを示す．基本的にはバイグラムの異なり数の利得が最大となる文を逐次的に選択していくだけである．

3.3 ラグランジュ緩和と制限長さの段階的緩和の導入

最終的に求めたい解は，重要な情報を最大限被覆し，かつ，バイグラム異なり数を最大とする要約である．よって，L_{max} による文字数制限と，それぞれの解の一致制約 $\boldsymbol{y} = \boldsymbol{z}$ のもとで以下の式を最大化すれば良い．

$$F(\boldsymbol{y}, \boldsymbol{z}) = f_{\text{knp}}(\boldsymbol{y}) + f_{\text{grd}}(\boldsymbol{z})$$

$\boldsymbol{y}$，$\boldsymbol{z}$ は長さ N のバイナリベクトルであり，文 i が要約に含まれる場合に i 番目の要素が 1 を取る．関数 f はナップサック問題，劣モジュラ問題で要約を評価した際のスコアを表す．このままでは，問題を解くことは困難なので，以下のラグラジュ双対問題を考える．

$$\min_{\boldsymbol{u}} \quad L(\boldsymbol{y}, \boldsymbol{z}, \boldsymbol{u}) \quad = F(\boldsymbol{y}, \boldsymbol{z}) + \sum_{i=1}^{N} u_i(y_i \quad z_i)$$

$\boldsymbol{u}$ はラグランジュ乗数であり，u_i は i 番目の文に対するペナルティを表す．この問題は Algorithm 3 の手続きにより解くことができる． $^{(k)}$ は，k 番目のラウンドにおけるペナルティを調整するパラメータである．以下の実験では $^{(k)} = 0.5/k$ とした．

提案法では，tf·idf に基づく単語重みと，バイグラムの異なり数による利得という，それぞれが出力する要約の傾向が大きく異なる問題を組合せている．このため，上記の手続きだけでは現実的なラウンド数で収束せず，合意を形成できない場合がある．

文献 [Rush 10, Chang 11] では，収束しない場合の方策として，制約を追加してラグランジュ緩和を厳しくするというアプローチが採られている．しかし，このアプローチは，制約の追加に連れて，動的計画法において記憶する必要がある状態数が爆発的に増大するという問題がある．

Algorithm 4 要約生成アルゴリズム (長さ緩和あり)

Input:
  $D = \{s_1, ..., s_N\}$, L_{max}
Initialize:
  $u_i^{(0)} \leftarrow 0$ **for** $i \in \{1, \ldots, N\}$
  $L'_{max} \leftarrow L_{max}$
for $k \in \{1, \ldots, K\}$ **do**
  $\boldsymbol{y}^{(k)} \leftarrow \underset{\boldsymbol{y} \in \mathcal{Y}}{\text{argmax}}(f_{\text{knp}}(\boldsymbol{y}) + \sum_{i=1}^{N} u_i^{(k-1)} y_i)$
  $\boldsymbol{z}^{(k)} \leftarrow \underset{\boldsymbol{z} \in \mathcal{Z}}{\text{argmax}}(f_{\text{grd}}(\boldsymbol{z}) \quad \sum_{i=1}^{N} u_i^{(k-1)} z_i)$
  if check_converge($\boldsymbol{y}^{(k)}, \boldsymbol{z}^{(k)}$)
    return intersection($\boldsymbol{y}^{(k)}, \boldsymbol{z}^{(k)}$)
  elif match_count($\boldsymbol{y}^{(k)}, \boldsymbol{z}^{(k)}$) $> L_{max}$
    $L'_{max} \leftarrow (1 + \frac{(\beta-1)}{2}) L'_{max}$
  else
    for $i \in \{1, \ldots, N\}$ **do**
      $u_i^{(k)} \leftarrow u_i^{(k-1)} \quad \alpha^{(k)}(y_i^{(k)} \quad z_i^{(k)})$
  if $k \bmod R = 0$
    $L'_{max} \leftarrow \beta L'_{max}$
return $\boldsymbol{y}^{(K)}$

我々は，要約生成は長さをパラメータとして持つことに着目し，収束性を高めるためのヒューリスティクとして，段階的な長さの緩和を導入する．具体的には，一定ラウンド数を過ぎても合意を形成できない場合，その度に制限長さを段階的に延ばし，ナップサック問題と劣モジュラ最大化問題双方から出力された文のうち，一致している文のみで合意を形成するよう変更する．ここで，一致している文のみでの合意とは，一致している文の長さの合計が制限長さ L_{max} を越えず，しかも，一致していない文のうち，どの文を足したとしても L_{max} を越える場合とした．長さの緩和を導入後の手続きを Algorithm 4 に示す．出力したすべての文集合の過不足ない一致に代わって，出力のうち一部についての一致で良いとすることで，合意が形成されやすくなると考える．

4 評価実験

評価実験には TSC 3 の複数文書要約データを利用した．トピック数は 30 なので，30 トピックでの ROUGE-1 スコア [Lin 04] の平均で評価した．

比較に用いた手法は，合意解を得る前の 2 手法である tf·idf の単語重みを用いてナップサック問題として解いた手法と, バイグラムによる貪欲法として解いた手法, および [McDonald 07] 同様，文の類似度をペナ

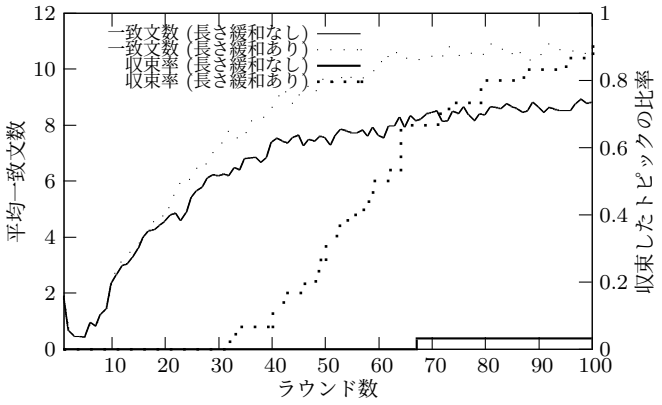

図 1: ラウンド数と一致した文数の比率，収束したトピックの比率の関係

表 1: 各手法による ROUGE-1 スコア

手法	ROUGE-1
提案法 (長さ緩和なし)	0.478
提案法 (長さ緩和あり)	0.484
tf·idf によるナップサック	0.461
バイグラムによる貪欲法	0.344
動的計画法へ類似度ペナルティを導入	0.474

ルティとして与えて動的計画法のみで解く方法の 3 種類である．

最大ラウンド数 K は 100 とした．また，長さ制限の緩和を行う場合のパラメータ R, $\quad$ の値は，それぞれ 10, 1.1 とた．

実験の結果を表 1 に示す．表より，提案手法が合意を取る前の 2 手法と比較して ROUGE-1 スコアが良く，異なる 2 手法の間で合意解を形成することの有効性を示している．また，McDonald の手法よりも ROUGE-1 スコアが高いことから，冗長な情報が少なく重要な情報を被覆するという目的を達成できたと考える．長さ緩和を導入することで成績はさらに向上しており，その有効性も確認できた．これは長さ緩和を導入することで 2 つの手法の間での合意解を得やすくなったことが主な要因であると考える．長さ緩和導入前後のラウンド数と合意した文の数，収束したトピックの比率の関係を図 1 に示す．図より，長さ緩和を導入することで，合意できた文の数が一貫して増えている．さらに，収束するトピックの数も 1/30 に対して，27/30 とほとんどのトピックで合意解を得ることができた．

5 おわりに

本稿では，重要情報を網羅するという目的を厳密解を得ることのできるナップサック問題，情報の重複を最小化するという問題を近似が必要な劣モジュラ関数最大化問題として捉え，それらが生成した要約の合意解を最終的な要約とする要約モデルを提案した．合意解を得るためラグランジュ緩和を導入したが，これを単純に用いるだけでは合意解を得ることが難しいことを実験的に示し，それを解決する方策として，長さ制限の段階的緩和を導入した．TSC-3 のデータを用いて評価実験を行ったところ，合意解を得る前よりも ROUGE スコアが向上することを確認し，長さ緩和を導入することでさらにスコアが向上することを確認した．また，解の収束性が改善されることも確認した．

参考文献

[Chang 11] Chang, Y.-W. and Collins, M.: Exact Decoding Phrase-based Translation Model through Lagrangian Relaxation, in *Proceedings of EMNLP* (2011)

[Filatova 04] Filatova, E. and Hatzivassiloglou, V.: A Formal Model for Information Selection in Multi-Sentence Sentece Extraction, in *Proceedings of the 20th COLING* (2004)

[Goldstein 00] Goldstein, J., Mittal, V., Carbonell, J., and Kantrowitz, M.: Multi-Document Summarization by Sentence Extraction, in *Proceedings of ANLP/NAACL Workshop on Automatic Summarization* (2000)

[Hirao 04] Hirao, T., Okumura, M., Fukushima, T., and Nanba, H.: Text Summarization Challenge 3, in *Proc. of the NTCIR-4*, pp. 407–411 (2004)

[Lin 04] Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries, in *Proc. of Workshop on Text Summarization Branches Out*, pp. 74–81 (2004)

[Lin 10] Lin, H. and Bilmes, J.: Multi-document Summarization via Budget Maximization of Submodular Functions, in *Proceedings of the NAACL/HLT-2010* (2010)

[Lin 11] Lin, H. and Bilmes, J.: A Class of Submodular Functions for Document Summarization, in *Proceedings of the ACL/HLT* (2011)

[McDonald 07] McDonald, R.: A Study of Grobal Inference Algorithm in Multi-document Summarization, in *Proceedings of the 29th ECIR* (2007)

[Rush 10] Rush, A. M., Sontag, D., and Collins, M.: On Dual Decomposition and Linear Programming for Natural Language Processsing, in *Proceedings of EMNLP* (2010)

[Takamura 09] Takamura, H. and Okumura, M.: Text Summarization Model based on Maximum Coverage Problem and its Variant, in *Proceedings of the 14th EACL* (2009)